



Quaternionic Sparse Approximation

Quentin Barthélemy, Anthony Larue, Jerome I. Mars

► To cite this version:

Quentin Barthélemy, Anthony Larue, Jerome I. Mars. Quaternionic Sparse Approximation. AGACSE 2012 - 5th conference on Applied Geometric Algebras in Computer Science and Engineering, Jul 2012, La Rochelle, France. Paper#4, p1-11. hal-00730188

HAL Id: hal-00730188

<https://hal.science/hal-00730188>

Submitted on 7 Sep 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Quaternionic Sparse Approximation

Quentin Barthélemy, Anthony Larue and Jérôme I. Mars

Abstract In this paper, we introduce a new processing procedure for quaternionic signals through consideration of the well-known orthogonal matching pursuit (OMP), which provides sparse approximation. We present a quaternionic extension, the quaternionic OMP, that can be used to process a right-multiplication linear combination of quaternionic signals. As validation, this quaternionic OMP is applied to simulated data. Deconvolution is carried out and presented here with a new spikegram that is designed for visualization of quaternionic coefficients, and finally this is compared to multivariate OMP.

1 Introduction

In signal processing, some tools have been recently extended to the quaternion space \mathbb{H} . For example, we can cite quaternionic correlation for vector images [8], quaternionic adaptive filtering [11], quaternionic independent component analysis [13], and blind extraction of quaternionic sources [5].

For the sparsity domain, sparse approximation algorithms [12] are given for real or complex signals. To our knowledge, these have not been applied to quaternionic signals. Considering orthogonal matching pursuit (OMP) [9], we present an extension to quaternions. This algorithm, which is termed quaternionic OMP (Q-OMP), can be used to process quadrivariate signals (and thus including trivariate signals).

Quentin Barthélemy
CEA-LIST, PC 72, 91191 Gif-sur-Yvette Cedex, France, e-mail: quentin.barthelemy@cea.fr

Anthony Larue
CEA-LIST, PC 72, 91191 Gif-sur-Yvette Cedex, France, e-mail: anthony.larue@cea.fr

Jérôme I. Mars
GISPA-Lab, 11 rue des mathématiques BP46, 38402 Saint Martin d'Hères, France, e-mail: jerome.mars@gipsa-lab.grenoble-inp.fr

For the quaternionic signal $y \in \mathbb{H}^N$ of N samples and a dictionary $\Phi \in \mathbb{H}^{N \times M}$ of M atoms $\{\phi_m\}_{m=1}^M$, the decomposition of the signal y is carried out on the dictionary Φ such that:

$$y = \sum_{m=1}^M \phi_m x_m + \varepsilon, \quad (1)$$

assuming $x_m \in \mathbb{H}$ are the coding coefficients and $\varepsilon \in \mathbb{H}^N$ the residual error. It is this right-multiplication model that will be considered for the following quaternionic sparse approximation. This model is used in different real-world applications, such as wind forecasting and colored images denoising [11], and in blind source extraction of EEG mixtures [5].

In this paper, we first consider sparse approximation and the OMP algorithm in Section 2. We then present the quaternionic extension Q-OMP in Section 3. In Section 4, we specify our work for the shift-invariant case, and we introduce a new visualization tool for quaternionic sparse decompositions. Finally, in Section 5, the Q-OMP is applied to deconvolute simulation data, and then compared to multivariate OMP (M-OMP).

2 Sparse Approximation

The sparse approximation principle and the OMP algorithm are presented in this section, with processing of only complex signals.

2.1 Principle and existing algorithms

Considering a signal $y \in \mathbb{C}^N$ of N samples and a dictionary $\Phi \in \mathbb{C}^{N \times M}$ of M atoms $\{\phi_m\}_{m=1}^M$, the decomposition of the signal y is carried out on the dictionary Φ such that:

$$y = \Phi x + \varepsilon, \quad (2)$$

assuming $x \in \mathbb{C}^M$ are the coding coefficients and $\varepsilon \in \mathbb{C}^N$ the residual error. The dictionary is normed, which means that its columns (atoms) are normed, so that coefficients x reflect the energy of each atom in the signal. Moreover, the dictionary is said to be redundant when $M > N$.

One way to formalize the decomposition under the sparsity constraint is:

$$\min_x \|y - \Phi x\|_2^2 \text{ s.t. } \|x\|_0 \leq K, \quad (3)$$

where $K \ll M$ is a constant and $\|x\|_0$ the ℓ_0 pseudo-norm that is defined as the cardinality of x . This formulation is composed of a data-fitting term and a term of sparsification, to obtain the sparsest vector x . Pursuit algorithms [12] tackle sequentially (3) increasing K iteratively, although unfortunately this optimization is nonconvex:

that means the obtained solution can get stuck in a local minimum. Nevertheless, these algorithms are fast when searching very few coefficients. Among the multiple ℓ_0 -Pursuit algorithms, we can cite the well-known matching pursuit (MP) [6], its orthogonal version, OMP [9] and multivariate OMP (M-OMP) [1] for treating multivariate signals. Note that another way consists of relaxing the sparsification term from an ℓ_0 norm to an ℓ_1 norm, which gives a convex optimization problem [12].

2.2 Review of Orthogonal Matching Pursuit

We present the step-by-step OMP that is introduced in [9] with complex signals. Given a redundant dictionary Φ , OMP produces a sparse approximation of a signal y (Algorithm 1).

After an initialization (step 1), OMP selects at the current iteration k the atom that produces the absolute strongest decrease in the mean square error (MSE) $\|\varepsilon^{k-1}\|_2^2$. This is equivalent to selecting the atom that is the most correlated with the residue. The inner products between the residue ε^{k-1} and atoms ϕ_m are computed (step 4). The selection (step 6) searches the maximum of their absolute values to determine the optimal atom ϕ_{m^k} , denoted d_k . An active dictionary $D^k \in \mathbb{C}^{N \times k}$ is formed, which collects all of the selected atoms (step 7). Coding coefficients x^k are computed via the orthogonal projection of y on D^k (step 8). This is often carried out recursively by different methods using the current correlation value $C_{m^k}^k$: QR factorization [3], Cholesky factorization [2], or block matrix inversion [9]. The obtained coefficients vector $x^k = [x_{m^1}; x_{m^2} \dots x_{m^k}]^T$ is reduced to its active (*i.e.* nonzero) coefficients, with $(\cdot)^T$ denoting the transpose operator.

Different stopping criteria (step 11) can be used: a threshold on k for the number of iterations, a threshold on the relative MSE (rMSE) $\|\varepsilon^k\|_2^2 / \|y\|_2^2$, or a threshold on the decrease in the rMSE. At the end, the OMP provides a K -sparse approximation of y :

$$\hat{y}^K = \sum_{k=1}^K x_{m^k} \phi_{m^k}. \quad (4)$$

Used thereafter, M-OMP [1] deals with the multivariate signals acquired simultaneously.

3 Quaternionic Orthogonal Matching Pursuit

In this section, we present the Q-OMP, the quaternionic extension of the OMP. As mentioned above, different implementations of the OMP projection step exist. In the following, we have chosen to extend the block matrix inversion method [9]. We first outline the quaternionic space and notations, and we then detail the Q-OMP algorithm.

Algorithm 1 : $x = \text{OMP}(y, \Phi)$

```

1: initialization :  $k = 1$ ,  $\varepsilon^0 = y$ , dictionary  $D^0 = \emptyset$ 
2: repeat
3:   for  $m \leftarrow 1, M$  do
4:     Inner Products :  $C_m^k \leftarrow \langle \varepsilon^{k-1}, \phi_m \rangle$ 
5:   end for
6:   Selection :  $m^k \leftarrow \arg \max_m |C_m^k|$ 
7:   Active Dictionary :  $D^k \leftarrow D^{k-1} \cup \phi_{m^k}$ 
8:   Active Coefficients :  $x^k \leftarrow \arg \min_x \|y - D^k x\|_2^2$ 
9:   Residue :  $\varepsilon^k \leftarrow y - D^k x^k$ 
10:   $k \leftarrow k + 1$ 
11: until stopping criterion

```

3.1 Quaternions

The quaternionic space, denoted as \mathbb{H} , is an extension of the complex space \mathbb{C} using three imaginary parts [4]. A quaternion $q \in \mathbb{H}$ is defined as: $q = q_a + q_b i + q_c j + q_d k$, with $q_a, q_b, q_c, q_d \in \mathbb{R}$ and with the imaginary units defined as: $ij = k$, $jk = i$, $ki = j$ and $ijk = i^2 = j^2 = k^2 = -1$. The quaternionic space is characterized by its noncommutativity: $q_1 q_2 \neq q_2 q_1$. The scalar part is $S(q) = q_a$, and the vectorial part is $V(q) = q_b i + q_c j + q_d k$. If its scalar part is null, a quaternion is said to be pure and full otherwise. The conjugate q^* is defined as: $q^* = S(q) - V(q)$ and we have $(q_1 q_2)^* = q_2^* q_1^*$.

Now considering quaternionic vectors $q_1, q_2 \in \mathbb{H}^N$, we define the inner product as: $\langle q_1, q_2 \rangle = q_2^H q_1$, with $(\cdot)^H$ denoting the conjugate transpose operator. The associated ℓ_2 norm is denoted by $\|\cdot\|_2$. Note that an alternative definition can be chosen: $\langle q_1, q_2 \rangle = q_1^H q_2$, which is only the conjugate of the previous one.¹

In the previous section, OMP was explained for complex variables with the traditional left-multiplication between scalars and vectors. Due to the noncommutativity of quaternions, only the right-multiplication given in Eq. (1) will be considered in the following. Moreover, the quadrivariate data studied are filled in the full quaternionic variables thereafter.

3.2 Algorithm description

Owing to noncommutativity, the variables order is now crucial. The description of the Q-OMP algorithm is similar to Algorithm 1 (we have taken care to already give the appropriate parameter order in the OMP algorithm), assuming quaternionic signals and the use of right-multiplication. In this section, the changes are detailed, and in particular, the orthogonal projection.

¹ Note also that the inner product in $\mathbb{R}^{N \times 4}$: $\langle q_1, q_2 \rangle = S(q_1^H q_2)$, which is often used for quaternionic processings, is not considered here.

The inner product $\langle \varepsilon^{k-1}, \phi_m \rangle$ (step 4) remains the expression to maximize to select the optimal atom (see Appendix). It is now the quaternionic inner product defined in Section 3.1. Coefficients x^k (step 8) are calculated by orthogonal projection: the recursive procedure [9] is extended to quaternions and with right-multiplication, as is described below (for normed kernels). Foremost, A_k is defined as the Gram matrix of the active dictionary D^{k-1} :

$$A_k = (D^{k-1})^H D^{k-1} = \begin{bmatrix} \langle d_1, d_1 \rangle & \langle d_2, d_1 \rangle & \dots & \langle d_{k-1}, d_1 \rangle \\ \langle d_1, d_2 \rangle & \langle d_2, d_2 \rangle & \dots & \langle d_{k-1}, d_2 \rangle \\ \vdots & \vdots & \ddots & \vdots \\ \langle d_1, d_{k-1} \rangle & \langle d_2, d_{k-1} \rangle & \dots & \langle d_{k-1}, d_{k-1} \rangle \end{bmatrix}. \quad (5)$$

At iteration k , the recursive procedure for the orthogonal projection is computed in seven stages:

- 1: $v_k = (D^{k-1})^H d_k = [\langle d_k, d_1 \rangle; \langle d_k, d_2 \rangle \dots \langle d_k, d_{k-1} \rangle]^T$,
- 2: $b_k = A_k^{-1} v_k$,
- 3: $\beta = 1/(\|d_k\|^2 - v_k^H b_k) = 1/(1 - v_k^H b_k)$,
- 4: $\alpha_k = C_{m^k}^k \cdot \beta$.

To provide the orthogonal projection, coefficients x_{m^i} of vector x^k are corrected at each iteration. Adding a superscript (to coefficients x_{m^i}) denotes the iteration, and the update is:

- 5: $x_{m^i}^k = x_{m^i}^{k-1} - b_k \alpha_k$, for $i = 1 \dots k-1$,
- 6: $x_{m^k}^k = \alpha_k$.

The Gram matrix update is given by:

$$A_{k+1} = \left[\begin{array}{c|c} A_k & v_k \\ \hline v_k^H & 1 \end{array} \right] \quad (6)$$

and using the block matrix inversion formula, we obtain its left-inverse:

$$7: \quad A_{k+1}^{-1} = \left[\begin{array}{c|c} A_k^{-1} + \beta b_k b_k^H & -\beta b_k \\ \hline -\beta b_k^H & \beta \end{array} \right]. \quad (7)$$

For the first iteration, the procedure is reduced to: $x_{m^1}^1 = C_{m^1}^1$ and $A_1 = 1$.

As Algorithm 1, and with the described modifications, the Q-OMP provides a K -sparse approximation of y :

$$\hat{y}^K = \sum_{k=1}^K \phi_{m^k} x_{m^k}. \quad (8)$$

4 The shift-invariant case and the spikegram

In this section, we focus on the shift-invariant case, and a new spikegram for quaternionic decompositions is introduced.

4.1 The shift-invariant case

In the shift-invariant case, we want to sparsely code the signal y as a sum of a few short structures, named kernels, that are characterized independently of their positions. This is usually applied to time series data, and this model avoids the block effects in the analysis of largely periodic signals, and provides a compact kernel dictionary [10].

The L shiftable kernels of the compact dictionary Ψ are replicated at all of the positions, to provide the M atoms of the dictionary Φ . The N samples of the signal y , the residue ε , and the atoms ϕ_m are indexed ² by t . The kernels $\{\psi_l\}_{l=1}^L$ can have different lengths. The kernel $\psi_l(t)$ is shifted in τ samples to generate the atom $\psi_l(t - \tau)$: zero-padding is carried out to have N samples. The subset σ_l collects the active translations τ of the kernel $\psi_l(t)$. For the few kernels that generate all of the atoms, Eq. (2) becomes:

$$y(t) = \sum_{m=1}^M \phi_m(t) x_m + \varepsilon(t) = \sum_{l=1}^L \sum_{\tau \in \sigma_l} \psi_l(t - \tau) x_{l,\tau} + \varepsilon(t). \quad (9)$$

To sum up, in the shift-invariant case, the signal y is approximated as a weighted sum of a few shiftable kernels ψ_l .

The Q-OMP algorithm is now specified for the shift-invariant case. The inner product between the residue ε^{k-1} and each atom ϕ_m (step 4) is now replaced by the correlation with each kernel ψ_l . Quaternionic correlation is defined in [8], although with the alternative inner product. In our case, the non-circular quaternionic correlation between quaternionic signals $q_1(t)$ and $q_2(t)$ is:

$$\Gamma \{q_1, q_2\}(\tau) = \langle q_1(t), q_2(t - \tau) \rangle = q_2^H(t - \tau) q_1(t). \quad (10)$$

The selection (step 6) determines the optimal atom that is now characterized by its kernel index l^k and its position τ^k . The orthogonal projection (step 8) gives the vector $x^k = [x_{l^1, \tau^1}; x_{l^2, \tau^2} \dots x_{l^k, \tau^k}]^T$. Finally, Eq. (8) of the K -sparse approximation becomes:

$$\hat{y}^K = \sum_{k=1}^K \psi_{l^k}(t - \tau^k) x_{l^k, \tau^k}. \quad (11)$$

² Note that $a(t)$ and $a(t - t_0)$ do not represent samples, but the signal a and its translation of t_0 samples.

4.2 The spikegram for quaternionic decompositions

We now explain how to visualize the coefficients obtained from a shift-invariant quaternionic decomposition. Usually, real coding coefficients $x_{l,\tau}$ are displayed by a time-kernel representation called a spikegram [10]. This condenses three indications:

- the temporal position τ (abscissa),
- the kernel index l (ordinate),
- the coefficient amplitude $x_{l,\tau}$ (gray level of the spike).

This presentation allows an intuitive readability of the decomposition. With complex coefficients, the coefficient modulus is used for the amplitude, and its argument gives the angle, which is written next to the spike [1]. This coefficient presentation provides clear visualization.

To display quaternionic coefficients and to maintain good visualization, each quaternionic coefficient is written such that:

$$x_{l,\tau} = |x_{l,\tau}| \cdot q_{l,\tau} \quad \text{and} \quad q_{l,\tau} = e^{i\theta_{l,\tau}^1} \cdot e^{k\theta_{l,\tau}^2} \cdot e^{j\theta_{l,\tau}^3}, \quad (12)$$

with the coefficient modulus $|x_{l,\tau}|$ that represents the atom energy, and $q_{l,\tau}$ as a unit quaternion (*i.e.* its modulus is equal to 1). This unit quaternion has only 3 degrees of freedom, which we arbitrary define as the Euler angles [4]. These parameters describe in a univocal way the considered quaternion on the unit sphere. Thereafter, we use this practical angle formalism, although without any rotation in the processing.

Two color bars are set up for the quaternionic spikegram: one for coefficient amplitude, and the other for the parameters assimilated to the Euler angles. The angles scale, defined from -180 to 180 in degrees, is visually circular; a negative value just above -180 thus appears visually close to a positive value just below 180. Finally, the quaternionic coefficients $x_{l,\tau}$ are displayed in this way with six indications:

- the temporal position τ (abscissa),
- the kernel index l (ordinate),
- the coefficient amplitude $|x_{l,\tau}|$ (color bar),
- the 3 parameters $\theta_{l,\tau}^1, \theta_{l,\tau}^2, \theta_{l,\tau}^3$ displayed vertically (circular color bar).

This representation is used for Fig. 1 and 2, and it provides an intuitive visualization of the different parameters.

5 Experiments and Comparisons

In this section, the Q-OMP is illustrated with deconvolution and then compared to the M-OMP. In this experiment, the data considered are trivariate, rather than quadrivariate, only so as not to load down figures and to maintain clearer reading.

Filled in pure quaternions, trivariate signals are processed using full quaternions for coding coefficients. A dictionary Ψ of $L=6$ non-orthogonal kernels is artificially built, and five coding coefficients $x_{l,\tau}$ are generated (with overlaps between atoms). First, the quaternionic signal $y \in \mathbb{H}^N$ is formed using Eq. (11), which is plotted in Fig. 1a. The first imaginary part y_b is plotted as the solid green line, the second, y_c , as the dotted black line, and the third, y_d , as the dashed blue line. Then, white Gaussian noise is added, giving the noised signal y_n that is now characterized by an RSB of 0 dB. This is shown in Fig. 1b, maintaining the the line style convention. Then, we deconvolute this signal y_n through the dictionary Ψ using Q-OMP with $K=5$ iterations. The denoised signal \hat{y}_n , that is obtained by computing the K -sparse approximation of y_n , is plotted in Fig. 1c. The coding coefficients $x_{l,\tau}$ are the result of the deconvolution, and they are shown in Fig. 1d, using the spikegram introduced in Section 4.2.

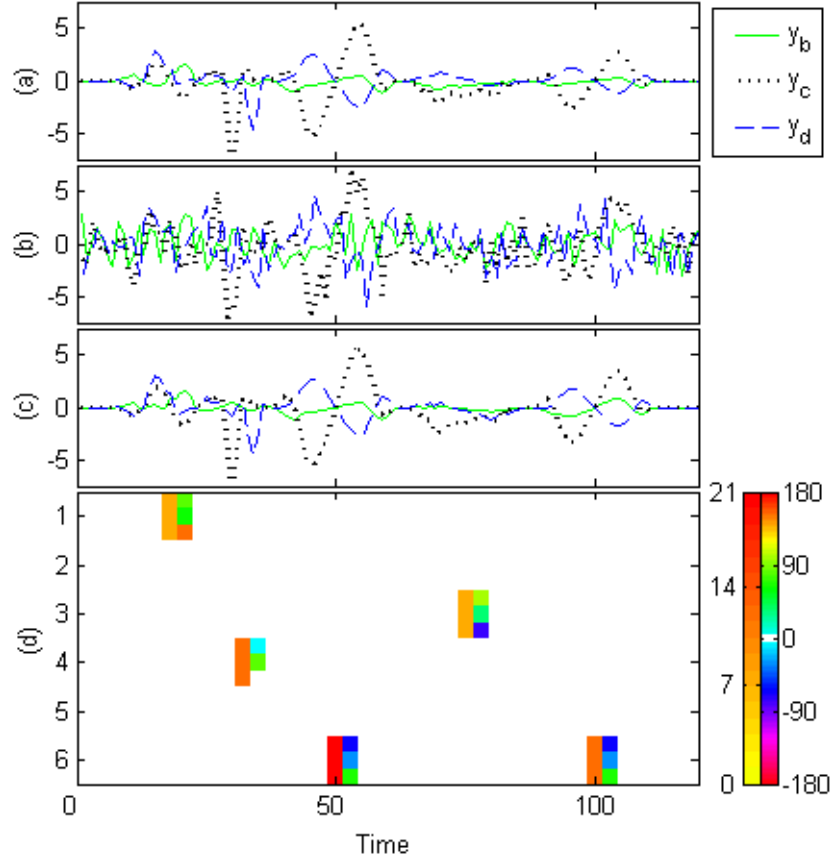


Fig. 1 Original (a), noised (b) and approximated (c) quaternionic signals (first imaginary part y_b as the solid green line; the second, y_c , as the dotted black line; and the third, y_d , as the dashed blue line), and the associated spikegram (d).

We observe that Q-OMP recovers the generated coefficients well, and the approximation \hat{y}_n is close to the original signal y ; the rMSE is only 2.8 %. This experiment is randomly repeated 100 times, and the averaged rMSE is 4.7%. This illustrates the Q-OMP efficiency for denoising and deconvolution. In Fig. 1d, note that coefficients $x_{6,50}$ and $x_{6,100}$ are coded with different amplitudes, but with the similar unit quaternion q .

Q-OMP is now compared to M-OMP, only using the trivariate case. The pure quaternionic signal y_n is now filled in a trivariate real signal $\underline{y}_n \in \mathbb{R}^{N \times 3}$ as well as the kernel dictionary. The M-OMP is applied with $K = 5$ iterations, and this gives the denoised signal $\hat{\underline{y}}_n$ that is plotted in Fig. 2a. The rMSE is 81.7%, and the average over 100 experiments is 76.8%. The associated spikegram is shown in Fig. 2b, using the original visualization. We observe that the strong coefficients are relatively well recovered, although the others are not (temporal shift τ , kernel index l , and amplitude). However, although the strongest coefficients are recognized, this is not sufficient to obtain a satisfactory approximation. Indeed, multivariate sparse approximation is not adapted to this case, as it cannot take into account the cross-terms of the quaternionic vectorial part.

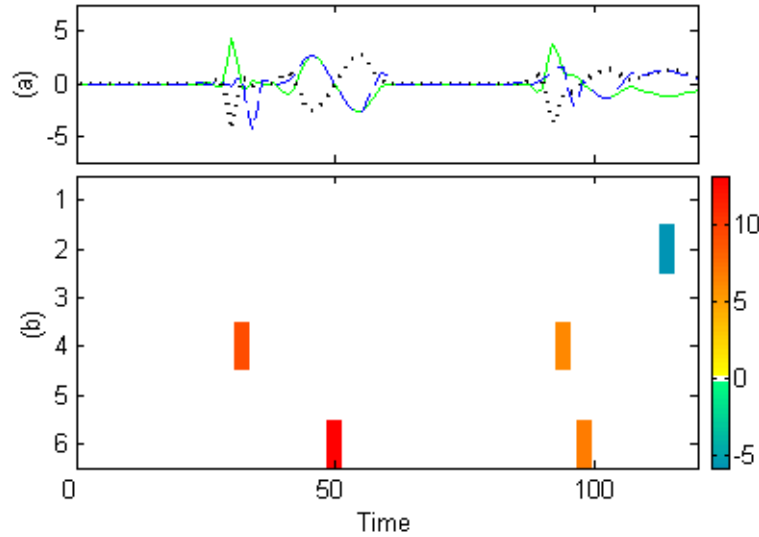


Fig. 2 Approximated trivariate real signal $\hat{\underline{y}}_n$ (a) and its associated spikegram (b).

A full quaternionic signal $y \in \mathbb{H}^N$ giving a quadrivariate real signal $\underline{y} \in \mathbb{R}^{N \times 4}$ is now considered. If the coefficients are strictly real, the two methods are equivalent. If not, the Q-OMP performs better, although for the complexity, the quadrivariate correlation only has 4 terms, whereas the quaternionic one has 16.

6 Conclusion

We have presented here a new tool for sparse approximation with quaternions: the Q-OMP, a quaternionic extension of the OMP. This processes a right-multiplication linear combination of quaternionic signals. We have also presented a new spikegram visualization for the quaternionic coefficients. For the validation, the Q-OMP was applied to deconvolute simulation data, and is compared to the M-OMP.

The potential uses of Q-OMP include quaternionic signal processing such as deconvolution, denoising, variable selection, dimensionality reduction, dictionary learning, and all of the other classical applications that are based on sparsity. Prospects are to present the left-multiplication Q-OMP.

Acknowledgements The authors thank N. Le Bihan from GIPSA-DIS and Prof. S. Sangwine from University of Essex for their precious advises about quaternions.

Appendix

The MSE objective function is $J = \|\varepsilon\|_2^2 = \varepsilon^H \varepsilon$. The derivation of J with respect to x is computed below.

To not lengthen the paper, calculus stages are not completely detailed.

$$\begin{aligned} \frac{\partial J}{\partial x} &= \frac{\partial J}{\partial x_a} + \frac{\partial J}{\partial x_b} i + \frac{\partial J}{\partial x_c} j + \frac{\partial J}{\partial x_d} k \\ &= \frac{\partial \varepsilon^H}{\partial x_a} \varepsilon + \varepsilon^H \frac{\partial \varepsilon}{\partial x_a} + \frac{\partial \varepsilon^H}{\partial x_b} \varepsilon i + \varepsilon^H \frac{\partial \varepsilon}{\partial x_b} i + \frac{\partial \varepsilon^H}{\partial x_c} \varepsilon j + \varepsilon^H \frac{\partial \varepsilon}{\partial x_c} j + \frac{\partial \varepsilon^H}{\partial x_d} \varepsilon k + \varepsilon^H \frac{\partial \varepsilon}{\partial x_d} k. \end{aligned} \quad (13)$$

Developing all the terms of $\varepsilon = y - \phi x$ and $\varepsilon^H = y^H - x^* \phi^H$, we obtain:

$$\begin{aligned} \partial \varepsilon / \partial x_a &= -(\phi_a + \phi_b i + \phi_c j + \phi_d k) = -\phi & \partial \varepsilon^H / \partial x_a &= -(\phi_a^T - \phi_b^T i - \phi_c^T j - \phi_d^T k) = -\phi^H \\ \partial \varepsilon / \partial x_b &= -(-\phi_b + \phi_a i + \phi_d j - \phi_c k) & \partial \varepsilon^H / \partial x_b &= -(-\phi_b^T - \phi_a^T i - \phi_d^T j + \phi_c^T k) \\ \partial \varepsilon / \partial x_c &= -(-\phi_c - \phi_d i + \phi_a j + \phi_b k) & \partial \varepsilon^H / \partial x_c &= -(-\phi_c^T + \phi_d^T i - \phi_a^T j - \phi_b^T k) \\ \partial \varepsilon / \partial x_d &= -(-\phi_d + \phi_c i - \phi_b j + \phi_a k) & \partial \varepsilon^H / \partial x_d &= -(-\phi_d^T - \phi_c^T i + \phi_b^T j - \phi_a^T k). \end{aligned}$$

Replacing these eight terms in Eq. (13), we have:

$$\begin{aligned} \frac{\partial J}{\partial x} &= -\phi^H \varepsilon - \varepsilon^H \phi \\ &\quad -(-\phi_b^T - \phi_a^T i - \phi_d^T j + \phi_c^T k)(-\varepsilon_b + \varepsilon_a i + \varepsilon_d j - \varepsilon_c k) \end{aligned} \quad (14)$$

$$\begin{aligned} &\quad -\varepsilon^H(-\phi_a - \phi_b i - \phi_c j - \phi_d k) \\ &\quad -(-\phi_c^T + \phi_d^T i - \phi_a^T j - \phi_b^T k)(-\varepsilon_c - \varepsilon_d i + \varepsilon_a j + \varepsilon_b k) \end{aligned} \quad (15)$$

$$\begin{aligned} &\quad -\varepsilon^H(-\phi_a - \phi_b i - \phi_c j - \phi_d k) \\ &\quad -(-\phi_d^T - \phi_c^T i + \phi_b^T j - \phi_a^T k)(-\varepsilon_d + \varepsilon_c i - \varepsilon_b j + \varepsilon_a k) \end{aligned} \quad (16)$$

$$\begin{aligned} &\quad -\varepsilon^H(-\phi_a - \phi_b i - \phi_c j - \phi_d k) \\ &= -\phi^H \varepsilon + 2\varepsilon^H \phi + (14) + (15) + (16). \end{aligned} \quad (17)$$

Developing the three terms (14), (15) and (16), adding and factorizing, we obtain:

$$(14) + (15) + (16) = -\phi^H \varepsilon - 2\varepsilon^H \phi. \quad (18)$$

With Eq. (17), we finally have:

$$\begin{aligned} \frac{\partial J}{\partial x} &= -\phi^H \varepsilon + 2\varepsilon^H \phi - \phi^H \varepsilon - 2\varepsilon^H \phi \\ &= -2\phi^H \varepsilon = -2\langle \varepsilon, \phi \rangle. \end{aligned} \quad (19)$$

Thus, we can conclude that the atom which produces the strongest decrease of the MSE $\|\varepsilon\|_2^2$ is the most correlated to the residue, as in the complex case.

Remark that this quaternion derivation has been done with the sum of componentwise gradients. It is called pseudogradient by Mandic *et al.* who propose a quaternion gradient operator in [7]. Using these new derivative rules, we obtain $\partial J / \partial x^* = -\phi^H \varepsilon + 1/2 \varepsilon^H \phi$. However, maximizing this expression does not give the optimal atom. It does not allow to recover known atoms in a simulated signal.

References

1. Q. Barthélemy, A. Larue, A. Mayoue, D. Mercier, and J.I. Mars. Shift & 2D rotation invariant sparse coding for multivariate signals. *IEEE Trans. on Signal Processing*, 60:1597–1611, 2012.
2. S.F. Cotter, R. Adler, R.D. Rao, and K. Kreutz-Delgado. Forward sequential algorithms for best basis selection. *Proc. IEEE Vision, Image and Signal Processing*, 146:235–244, 1999.
3. G. Davis, S. Mallat, and Z. Zhang. Adaptive time-frequency decompositions with matching pursuits. *Optical Engineering*, 33:2183–2191, 1994.
4. A.J. Hanson. *Visualizing quaternions*. Morgan-Kaufmann/Elsevier, 2006.
5. S. Javidi, C.C. Took, C. Jahanchahi, N. Le Bihan, and D.P. Mandic. Blind extraction of improper quaternion sources. In *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing ICASSP '11*, 2011.
6. S.G. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. *IEEE Trans. on Signal Processing*, 41:3397–3415, 1993.
7. D.P. Mandic, C. Jahanchahi, and C.C. Took. A quaternion gradient operator and its applications. *IEEE Signal Processing Letters*, 18(1):47–50, 2011.
8. C.E. Moxey, S.J. Sangwine, and T.A. Ell. Hypercomplex correlation techniques for vector images. *IEEE Trans. on Signal Processing*, 51:1941–1953, 2003.
9. Y.C. Pati, R. Rezaifar, and P.S. Krishnaprasad. Orthogonal Matching Pursuit: recursive function approximation with applications to wavelet decomposition. In *Proc. Asilomar Conf. on Signals, Systems and Comput.*, 1993.
10. E. Smith and M.S. Lewicki. Efficient coding of time-relative structure using spikes. *Neural Comput.*, 17:19–45, 2005.
11. C.C. Took and D.P. Mandic. Quaternion-valued stochastic gradient-based adaptive IIR filtering. *IEEE Trans. on Signal Processing*, 58:3895–3901, 2010.
12. J.A. Tropp and S.J. Wright. Computational methods for sparse solution of linear inverse problems. *Proc. of the IEEE*, 98:948–958, 2010.
13. J. Via, D.P. Palomar, L. Vielva, and I. Santamaria. Quaternion ICA from second-order statistics. *IEEE Trans. on Signal Processing*, 59:1586–1600, 2011.